

## **REMARKS**

**The Examiner rejected Claims 1-20 under 35 U.S.C. 112, first paragraph as failing to comply with the written description requirement. The above amendments remove the phrase on which this rejection is based, and hence, render this rejection moot.**

**Claims 1-4, 7-9, 14-29, 31-33 and 36-40 are rejected by the Examiner under 35 U.S.C. 103(a) as being unpatentable over US 6,907,557 to Perez, et al (hereafter "Perez") (incorporating by reference US 6,401,220 to Grey, et al (hereafter "Grey")) in view of US 6,449,741 to Organ, et al (hereafter "Organ"). Applicant submits that as currently amended these claims are not obvious in view of the cited references.**

With respect to Claims 1 and 21, the Examiner looks to Grey as teaching a variation point at which a function call instruction is inserted by a designer of the computer program to pass control to a user-defined variation function. The Examiner looks to Grey (col. 12, lines 41-53) and (col. 14, lines 52-65) as supporting this reading of Grey. The first passage teaches that the test sequences executed by the TestStand in Grey contain steps that call external code modules. The second passage teaches that the TestStand passes variables and properties to the user routines. The user, in this case, is the designer of the test sequence. The above amendments to Claims 1 and 21 make it clear that the user is different from the designer of the program.

The Examiner maintains that Grey teaches that the function call instruction passes control to the user-defined variation function when the variation point in the computer program is reached and cites Grey, (col. 13, lines 50-58 and col. 14, line 53 to col. 15, line 9) as supporting this reading of Grey. The first passage teaches that steps in the control program call code modules. However, there is no teaching of a variation of the measurement process being contained in one of those code modules, the variation having been defined by the user. The second passage refers to a browser dialog box that is provided during the editing phase of a program to provide the program designer with various expressions available on the TestStand for implementing computations or other functions supported by the TestStand.

There is no teaching that this browser is even available during the operation of the execution of the test sequence.

The Examiner looks to Perez as teaching that user is permitted to modify the measurement process through a user-defined variation function while preventing the user from modifying the measurement process through particular sequences. Applicant must disagree with the Examiner's reading of Perez. The cited passages in Perez teach that variations in a measurement sequence (the parent sequence) can be generated by creating child sequences that utilize common sequences from the parent and some variations that are particular to the particular child sequence. This is no different from duplicating a source program and then editing the source program. Perez teaches a scheme in which changes to the parent sequence are automatically provided to the child sequences so that the child sequences can be kept up to date without having to recreate the child sequences. To this end, Perez teaches that some of the parent sequences are not allowed to be modified without losing this ability for automatic updating. The editor provided by Perez requires the designer to override the locks if the designer wishes to create a child sequence that is not provided with the automatic update feature. There is no teaching that a child sequence that modified one of these protected sequences could not be created by a designer in the form of a child sequence to be run on a computer. Furthermore, it should be noted that there is no teaching that the parent sequence is executing on the machine and then calls a child sequence. The parent sequences are merely models for the child sequences. Once a child sequence is generated, it is the child sequence that runs, not the parent.

Claims 2 and 31 depend from claims 1 and 21 respectively and additionally require a servicing element that services an interface realized by the measurement process. The Examiner maintains that Grey (col. 13, lines 7-15) provides the additional teaching. Applicant must disagree. The cited passage states that the overall measurement system provides runtime interfaces to certain standard packages such as LabView. However, the passage does not teach that any module provided by the user utilizes such an interface.

In the final rejection dated 12/13/2007, the Examiner states that as the computer program of Grey provides interfaces to the user to "utilize and interact with the resulting user-defined variation function" in the process modification software module, that module itself

must have “some type of interface” that enables communication with the computer program-supplied interface. The measurement process in Grey is the test sequence executing on the TestStand. While the test sequences may call other modules, there is no teaching that such a module is bound to the calling program in a manner other than that of a conventional subroutine call. Such a subroutine requires no servicing element with it, since such routines are loaded at the same time as the process that calls them. Accordingly, there are additional grounds for allowing claims 2, 31, and the claims dependent therefrom.

Claim 7, which depends from claim 1 through claim 2, has been rewritten to make it clear that the interface has an identity which is determined by the user and passed into the measurement process. Claim 36 has a corresponding limitation. Applicant submits that Grey (col. 13, lines -30) teaches that the user can re-write part of the operating system to provide a user interface in place of the standard interfaces that come with the system taught in Grey. However, there is no teaching that the identity of the user interface is passed into the measurement process application. Furthermore, since Grey teaches that the interface is part of the measurement process, there is no need to pass the identity of the user function into the process of Grey. Hence, Applicant submits that there are additional grounds for allowing Claims 7 and 26.

Claims 19 and 20 depend from claim 1 and additionally require that each of a plurality of variation points in the computer program be associated with one of a plurality of user-defined functions in the process modification software module. Claim 38 likewise requires a plurality of user-generated variation functions. The Examiner points to Grey, col. 13, lines 16-25 as providing this teaching. Applicant submits that while this passage mentions the possibility of “multiple concurrent executions” and breakpoints, there is no teaching regarding the association of each of a plurality of variation points with one of a plurality of user-defined functions as specified in the claims. In this regard, it should be noted that a break point turns control of the program to a user, not to a user supplied function that was bound to the software in question.

In the final rejection dated 12/13/2007, the Examiner states that providing a program with a plurality of points where a user is permitted to enter information is conventional. Applicant submits that providing points where a user may enter information to a program

falls well short of providing points associated with user-defined functions in a user-generated software module. Hence, Applicant submits that there are additional grounds for allowing claims 19, 20, and 38.

**Claims 5, 6, 10-13, 34 and 35, as may best be understood by the Examiner, are rejected under 35 U.S.C. 103(a) as being unpatentable over Perez in view of Organ and further in view of US 2002/002 6514 to Ellis, et al (hereafter "Ellis"). Applicant submits that these claims, as amended above, are not obvious in view of the cited references.**

The Examiner states that the combination of Perez/Grey and Organ discloses all the limitations of the claims except for requiring that the measurement and process modification be carried out using two separate computers communicating using a Simple Object Access Protocol or Common Object Request Broker Architecture protocol. The Examiner looks to Ellis for the missing teachings.

As noted above with respect to claims 1 and 21, from which these claims depend, Applicant submits that the combination of Perez/Grey and Organ does not teach the limitation requiring providing a **user-generated** process modification software module comprising a **user-defined variation function** for causing the variation, and requiring associating the function call instruction with the user-defined variation function prior to execution of the measurement process. Ellis does not provide the missing teachings.

Hence, Applicant submits that the Examiner has failed to make a *prima facie* case for obviousness with respect to claims 5, 6, 10-13, 34 and 35.

Respectfully Submitted,



Calvin B. Ward  
Registration No. 30,896  
Date: March 6, 2008

Agilent Technologies, Inc.  
Legal Department, M/S DL429  
Intellectual Property Administration  
P.O. Box 7599  
Loveland, CO 80537-0599  
Telephone (925) 855-0413  
Telefax (925) 855-9214